

N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED
IN THE INTEREST OF MAKING AVAILABLE AS MUCH
INFORMATION AS POSSIBLE

Programming Input-Output Operations
in FORTRAN on Magnetic Tape

V. N. Gorokov

Translation of "Programmirovaniye na FOR-
TRANE operatsiy vvoda-vyvoda na magnitnuyu
lentu," Academy of Sciences USSR, Institute of Space
Research, Moscow, Report PR-362, 1977, pp 1-9

(NASA-TM-76231) PROGRAMMING INPUT-OUTPUT
OPERATIONS IN FORTRAN ON MAGNETIC TAPE
(National Aeronautics and Space
Administration) 10 p HC A02/MF A01 CSCL 09B

N81-27826

Unclas
G3/61 26769

| | | | |
|---|--|--|-----|
| 1. Report No. NASA TM 76231 | 2. Government Accession No. | 3. Recipient's Catalog No. | |
| 4. Title and Subtitle Programming Input-Output Operations in FORTRAN on Magnetic Tape | | 5. Report Date JUNE 1980 | |
| | | 6. Performing Organization Code | |
| 7. Author(s) V. N. Gorokov | | 8. Performing Organization Report No. | |
| | | 10. Work Unit No. | |
| 9. Performing Organization Name and Address SCITRAN Box 5456 Santa Barbara, CA 93108 | | 11. Contract or Grant No. NASW-3198 | |
| | | 13. Type of Report and Period Covered Translation | |
| 12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546 | | 14. Sponsoring Agency Code | |
| | | | |
| 15. Supplementary Notes Translation of "Programmirovaniye na FORTRAN operatsiy vvoda-vyvoda na magnitnuyu lentu," Academy of Sciences USSR, Institute of Space Research, Moscow, Report PR-362, 1977, pp 1-9 | | | |
| 16. Abstract A manual is given for using the subprogram MTCFF design- ed for carrying out input-output operations in FORTRAN on magnetic tape within the framework of the operating system DOS ES EVM [Disk Operating System of the Unified Series of Computers] By means of this subprogram it is possible to perform any input-output operations of interest to the programmer: re- cording and reading various control operations in the direct and reverse directions, even though for these purposes the facilities of the language FORTRAN itself are limited. The subprogram can also be used in programs written in other programming languages. | | | |
| 17. Key Words (Selected by Author(s)) | | 18. Distribution Statement Unclassified - Unlimited | |
| 19. Security Classif. (of this report) Unclassified | 20. Security Classif. (of this page) Unclassified | 21. No. of Pages 10 | 22. |

Programming Input-Output Operations in FORTRAN on Magnetic Tape

V. N. Gorokov

1. General Considerations

/3*

As a part of the DOS ES EVM FORTRAN-4 offers users an extensive capacity for carrying out input-output operations. However, the organization of input-output on magnetic tape has a number of disadvantages. In the first place, the physical recording does not exceed 260 bytes in size and may be accompanied by service information necessary for programs which accomplish input-output in FORTRAN. This complicates and may even preclude the possibility of an exchange of magnetic tapes between users making up programs in different languages. Secondly, there are practically no service operations such as rewinding tape for a needed file, a needed block, etc. In the third place, it is impossible to read in the reverse direction.

To facilitate programming input-output operations, the subprogram MTCFF is proposed. The subprogram is written in the ASSEMBLER language and occupies 224 bytes storage. The subprogram is called by the operator

CALL MTCFF (list of parameters)

The first parameter is the operation code enclosed between apostrophes. The following codes are admissible.

FRE: reading in the forward direction,
BRE: reading in the backward direction,
WRT: record,

ORIGINAL PAGE
OF POOR QUALITY

* Numbers in margins indicate foreign pagination.

FSR: one step forward per block,
 BSR: one step backward per block,
 FSF: one step forward per file,
 BSF: one step backward per file,
 WTM: recording of tape brand,
 ERG: erase segment
 REW: rewind tape to the beginning,
 RUN: rewind and load.

The second parameter is an unsigned integer or a four byte integer variable whose value indicates the number of a device taken from the specification SYSnnn by which it is necessary to perform the indicated operation. For example, the operator

CALL MTCFF('REW',6)

may be required in order to rewind to the beginning the magnetic tape found on the device SYS006. The recording of the remaining parameters depends on the type of operations to be performed.

2. Recording

A recording of one block on magnetic tape is accomplished by the operator.

CALL MTCFF('WRT',N,A,L[,&LBL]).

Here: A is the name of the first element which is recorded on the magnetic tape. It may be a simple variable or the name of a file with or without indices;

L is the length of the block in bytes. It can be an unsigned integer or a four-byte integer variable with a pre-assigned value. The length of the block can vary from 18 to 65535 bytes. It registers L bytes, beginning with the ad-

dress indicated by the parameter A ;

LBL is the label of the operator in the program to which control will be transferred, if during the recording the physical end of the tape is reached. The block will be completely recorded on the tape, and it may still be possible to record one or more short blocks signaling the end of the information on the tape.

3. Reading

Reading a block from a magnetic tape is accomplished by the operator

```
CALL MTCFF ('code' N, A, L [, &LBL1, &LBL2 ]).
```

Depending on the operation, the tape moves forward or backward to the following interblock interval. L determines the length of the block. When reading in the forward direction the first byte read is located by means of the address shown by the parameter A, and then in the increasing order of addresses. When reading in the reverse direction the byte read is located by means of the address A+L-1 and then in decreasing order of addresses. Thus the results of reading the same block in different directions are identical. The following block is not read, even if L is specified as greater than the length of the block. If L is less than the length of the block, L of the first bytes read are recorded in storage.

If the length of the block is unknown, it can be read by first assigning to the four-byte integer variable L the value 0. Then the entire block is read, and the variable L receives a value equal to the length of the block in bytes. During reading in the reverse direction, the first byte read is located by means of the address A-1, etc., in the order of decreasing addresses.

It must be kept in mind that there is no supervision over observance of the limits of files; therefore, if a block is not placed in the part of storage allocated for its processing, another part of the data or program may be erased. To prevent this from happening, the length of the block can be checked by placing L equal to -1. In this case the length of the block is transmitted to L, but the recording of the block itself in storage is prevented. The parameter A must be specified, although in this case it is meaningless. After checking the length of the block L, the user can resume reading the same block.

The parameters LBL1 and LBL2 may be omitted, but if the presence of at least one of them is necessary, then the second is required. LBL1 is the label of the operator in the program to which control will be transferred, if the tape index mark is passed. This may signal, for example, the end of the data. LBL2 is the label of the operator in the program to which control will be transferred if an irregularity in the data is discovered. During recording and reading the transmission of data is carried out without any transformations, analogous to the formatless input-output in FORTRAN.

4. The control operations

All the remaining operations mentioned in Section 1 belong to the control operations. They are performed by means of the operator

CALL MTCFF('KOD',N[,M][,&LBL]).

Here: M shows how many times it is necessary to carry out the indicated operation. If M is omitted, then it is assumed that M = 1;

LBL is the label of the operator in the program to which

control will be shifted if:

a) during the operations WTM and ERG the physical end of the tape is reached;

b) during the operations BSR and FSR the tape index mark is passed.

This parameter cannot be specified for other operations. If such a situation arises, performance of the operation is terminated, regardless of whether the parameter LBL is assigned and whether the parameter M has come to an end. The subprogram does not inform the user about the number of actually completed operations. If $M = 0$, then the indicated operation is performed until the situation described in a) and b) arises.

5. Processing errors in data

When errors appear in data (a failure in parity, a discrepancy in CRC, etc.), the operational system carries out actions, attempting to get rid of the source of the failure. If the error cannot be rectified, then the operator will be sent the message

ØP11A DATA CHECK,

to which the operator may respond: CANCEL -- remove the job -- or IGNORE -- disregard the error. In the second case, the malfunctioning block is read into storage and control is transferred to the operator in the program noted in the label of LBL2, if any. The user can initiate any actions regarding the processing of information in which there may be error.

In a subprogram there are provided two additional input points: MTCFS and MTCFP. If the user resorts to the first (MTCFS), then if a malfunction occurs and the attempts at elimination are unsuccessful, the operator receives the message

ØP111 DATA CHECK.

Interference by the operator is not required. The malfunction is ignored and processing continues. Control is shifted to the operator noted in the program with the label LBL2, if any. If the user resorts to the subprogram MTCFF, the programs for eliminating a malfunction are not called when it arises, no message is sent to the operator, and if the label LBL2 has been assigned, control is shifted to the corresponding operator in the program. The use of these input points is recommended, if it is known that there are irregular recordings on the tape, since the process of eliminating malfunctions and interference by the operator consume considerable time.

/8

6. Additional possibilities

The subprogram MTCFF makes it possible to carry out input-output operations on certain facilities of the system. The codes of these facilities are: 256 - SYSRDR, 257 - SYSIPT, 258 - SYSPCH and 259 - SYSLST. Any operations permitted for these facilities may be performed.

With the operator SYSLOG it is possible to carry out input-output for communication facilities by means of the subprogram. The code for SYSLOG is 260 and the codes of the operations are:

WRT -- record without carriage return,
WTO -- record with carriage return,
QSN -- request for reading from the keyboard.

These operations are analogous to the reading-recording operations, and for their performance these same parameters must be assigned.

When the operation QSN is performed, a shift is made to

the label LBL1 if the operator at the end of a set of text presses the button AH(annul), instead of the button KT(end of text), signalling the admission of an error. In this case it is necessary to repeat the operation in order to give the operator the opportunity to rectify the error:

25 CALL MTCFF('QSN',260,A,10,&25).

With the FORTRAN facilities it is impossible to execute input with SYSLOG.

7. Calling the subprogram MTCFF with programs in other programming languages

The subprogram MTCFF adheres to the standard system agreement about communications; therefore it is necessary that the calling program also use these agreements. For example, the subprogram transmits to a calling program in register 15 the return code 0, 4 or 8, depending on the type of conclusion involved in the operation, and the calling program can check this /9 code. The FORTRAN translator generates commands, carrying out such a check, if in the list of parameters of the operator CALL there are expressions of the type &LBL.

The parameters N, M, L must always be four-byte integers, and this must be carefully observed. For example, in PL/1, constants recorded by digits are of the type decimal fixed, but not binary fixed; therefore they must not be used as parameters.

8. Remarks about programming

ORIGINAL PAGE IS
OF POOR QUALITY

It must be kept in mind that the subprogram organizes input-output on the physical level; therefore there is no processing of tape labels. The labels VOL, HDR, EOF, etc., are

ordinary blocks which do not differ from recordings of data as far as the subprogram is concerned. The physical input-output system also does not organize combinations of input-output with a calculation in the program. Control of the calling program is returned only after the requested operation is completely finished. This can lead to a reduction in the productivity of the system in the case of installations without multiprogramming. Therefore, if in a program a very large input-output is provided for, languages must be used which make it possible to organize such a combination, e.g., ASSEMBLER, COBOL, PL/1, etc.